Chapter 9

The use of the LEGO MINDSTORMS® System in Modeling the Foraging Behavior and Strategies of Simple Animals

Marc Albrecht

Dept. of Biology
Univ. of Nebraska at Kearney
905 W. 25th St.
Kearney NE, 68849
albrechtm@unk.edu
(308) 865-8548

Marc Albrecht received his undergraduate degree in Biology from Denison University in Ohio in 1987. He next attended Miami University of Ohio, finishing his Master's degree in 1990. He moved west to the University of Oklahoma to work on his Ph.D. under Dr. Gotelli. Dr. Albrecht joined the UNK Biology Department in 1996. His interests include Geographical Information Systems, remote sensing, and animal movement.

Reprinted From: Albrecht, M. 2002. The use of the LEGO MINDSTORMS® system in modeling the foraging behavior and strategies of simple animals. Pages 145-154, in Tested studies for laboratory teaching, Volume 23 (M. A. O'Donnell, Editor). Proceedings of the 23rd Workshop/Conference of the Association for Biology Laboratory Education (ABLE), 392 pages.

Contents

Introduction

The LEGO MINDSTORMS® system provides a rich simulation environment for evolutionary adaptation and animal behavior. It includes both hardware and software that may be thought of as analogous to anatomical features and behavior respectively. Modification of these two components, though through directed evolution when done by students, demonstrates that small modifications in existing structures or instructions can produce large changes in the effectiveness of organisms in obtaining resources from their environment.

I see many advantages to this system. It is off-the-shelf which minimizes set-up and preparation time. Using a commercial product means that availability and technical help will not be a problem. There are other advantages to commercial products, the LEGO MINSTORMS website includes new challenges, information on new products, robot competitions, and the ability to post and trade programs with other users. This website is open to anyone and does not require any institutional resources updates or content (*http://www.MINDSTORMS.com*).

The MINDSTORMS package also introduces students to technology such as object-oriented computer programming, infrared data transmission, and basic mechanical principles such as gear ratios and levers. The ways these technologies and principles are used here make them comfortable for most students.

The MINDSTORMS system makes clear that form, function, and behavior of animals are inextricably linked. One of the attractive features of this product is the seamless integration of form and behavior. The software is written by students and downloaded to the robots. The programs must be customized to the design of each robot and its required tasks. In this lab I propose a foraging exercise, but many other scenarios are possible, such as obstacle course, positive or negative phototaxis, thigmotaxis, or trail following.

The robots are autonomous once they are programmed. This is the essence of individual-based modeling (IBM). The simulated organisms operate based only on their own internal instruction set using their individual abilities. Giving the students background on modeling in general, and on this type of model, seems to add to their understanding of the exercise. Such conceptual framework is valuable as some students anthropomorphize the behavior of even simple invertebrates. Experiencing the MINDSTORMS system allows students to see animals and their behavior from the 'inside out'. It becomes clear that a change in a single software setting (possibly conceptualized as an allele change) may change the behavior of the animal considerably.

Materials

Materials for this lab include enough individual LEGO MINDSTORMS® kits for each laboratory group. Computers (PC, not Macintosh) must be available that meet the requirements for the MINDSTORMS software. All components necessary for completing this exercise come with the kits (infrared transmitter, software, all hardware parts). Only the area must be built beforehand, and populated with resources. I recommend several layers of paper if the floor is not a solid white color.

Notes for the Instructor

It is important to provide instruction in how to use the components of the system. Regardless of the task you choose to have the robots complete, providing diagrams from the product literature greatly reduces the amount of time the students need to get a working robot running, down to a single busy lab period if necessary. Unless there is a specific reason for encouraging the students to build from the ground up, I recommend providing some basic assembly instructions and diagrams.

The same is true of the MINDSTORMS software. It is a simple object-oriented programming language. This means that students move block icons around on the screen and connect them together to build the program. There are blocks that start motors, stop motors, make beeps, watch for sensor triggers, and more. Enumerating them all here would make this lab write up quite long. I suggest that you explore the software and develop your own materials for explaining the software to your students. (This has the additional advantage of getting you to explore the software.) Students can save their programs on the hard drive and download up to five of them to the RCX – the computer brick that is the core of the MINDSTORMS system. The language has its limitations, but is remarkably flexible given its limited scope.

An alternative to providing material about the software is to allow more time, perhaps up to another full lab period, to let the students work through the tutorial that is included with the software. Let students use the LEGO MINDSTORMS literature that outlines the functions of each programming block. If you have the time, this option works well as the students get a deeper understanding of how animals may have 'hard-wired' responses to stimuli.

Attention should be paid to group dynamics in this exercise. Some groups may have technical expertise in object-oriented programming or LEGO® toys in general and work very quickly. Other groups may lack these talents or may not divide their tasks up efficiently or may all be working to solve one problem and not working on anything else. Because the goal is to have a functioning robot at the end of the time allotted your expertise and intervention in groups that are not progressing well is important. The more closely together in time the groups are completed, the better.

The operational components of the hardware include sensors for touch and light intensity. These sensors match very well the sensory capabilities of simple animals such as planaria. The placement and functioning of the sensors is important in the building of the robots. The 'testbench' function of the software – where the robot communicates back to the PC software the state of the sensors is particularly useful in solving problems. One typical issue is the reflectance values of the light sensor. This sensor has a light source and a photosensor built into a typical 2x3 LEGO® brick. The sensor has more discrimination the closer it is to the surface it is supposed to read. In addition, white paper does not often give a reading of 100 on the sensor as you might expect, but more like 50, where a dark desktop, if glossy, may give a reading of 35. These readings are different enough if the trigger values in the ‘sensor watcher’ programming block are set correctly, but this is something that needs to be adjusted.

The exercise given here -- foraging -- is just one of many possibilities. The more the students understand the ecological concepts they are supposed to build into the robots, the more confident they will be in building their machines.

The Arena

The typical arena I use is a 4 x 4 m square. I place 2.5 x 10 cm boards around the edge of the arena to stop robots from leaving the field. To simulate food resources, I tape black laser-printed dots and bars on the paper floor of the arena. I have used all three basic configurations in resources placement (clumped, random, overdispersed), and even some others, *e.g.*, much more in one corner of the area simulating a cline or elevational gradient. Some options are to try a bigger arena, ones that are not square, or perhaps longer time periods – it’s up to you! Other important details are also up to you: do you tell the students the basic distribution scheme of the resources or leave it unknown? Do the robots start out anywhere, or at a particular spot (this can be important)? A word of warning: as the students are optimizing the settings on the light sensor for detecting the food resources they need to have samples of exactly the same paper, tape, and laser printed dots as in the actual arena. If the students use different paper with pen dots, the reflectance values are not the same, and this can lead to disappointing results!

Expansion and Customization

The LEGO MINDSTORMS® system is capable of considerable customization and expansion if desired. The LEGO Group sells sensors for pH, rotation, and temperature. They also market sensors from DCP Microdevelopments Limited (*http://www.lego.com/dacta/hardware/dcpchart.asp*) that are significantly more accurate and can measure parameters not available from LEGO products. Circuit boards that add on to or replace those that come with the MINDSTORMS system are also available from various specialized vendors. In terms of software, there are sources for using Unix instead of the proprietary language used by the LEGO Group. In addition, the communication format can be overridden if desired. I expect that the LEGO Group will continue to upgrade and add to the product line itself. Two examples of this: 1) the current software and firmware package is much more stable and has a few additions as compared to the software 12 months ago, and 2) the VisionCommand USB digital camera has been produced by the LEGO Group. With a computer (probably a laptop), this camera can be used in conjunction with the MINDSTORMS system to detect motion, follow trials, and select objects of a particular color from the environment (a lab on color vision and fruit ripening suggests itself here).

Outline for Students

Objectives

Your task is to build a robot that will discover as many food sources as possible in a two-minute period in a 4 x 4 m arena.

Introduction

Animals strive to maximize their energy intake. This is true on a daily and an evolutionary basis. One of the basic problems facing animals is what strategy is best for maximizing energy intake. Is it better to race around quickly or go slowly? Is it best to turn often or rarely? Is it better to have a pattern or turn randomly? One could answer that species evolve whichever behavior is best for their environment. This assumes perfect and immediate adaptation, which in reality does not happen. Therefore, animals must change as the resources and the environment change. Your robots will not know what the future resource state is, so you must do your best to make a creature that covers as much ground as carefully as possible with the components you have – just as animals must!

Many closely related species have similar body parts and behaviors but use them differently – hawks and owls for instance. The LEGO MINDSTORMS® kits have the same components, but their arrangement in each robot, and the software – the behavior – of each robot will be different. You are not mimicking evolution directly here. You are applying directed evolution and intelligent design to solve the problem. Try to keep in mind that among animals the process of optimization would take place through the mechanisms of variation between individuals and natural selection. What you are modeling here directly is how various foraging strategies work for animals, how their sensors and locomotive parts work together, and how small differences in behavioral choices (what an animal considers 'light' vs. 'dark') can make large differences in how an animal moves through its environment. These changes are often controlled on a genetic level.

The animal you are modeling here is a simple one. It is not capable of learning, it has no memory of resources it has discovered or their pattern. It follows its programmed instructions literally as it searches the environment for energy. Hopefully, its body will hold together as it bangs into walls and possibly other animals.

Using this system has shown me that when we discuss evolution one aspect of the effects of natural selection on body structure we usually do not discuss is durability! I have seen the most effective robot with the best programming be defeated by a design flaw like a weak bumper that continually crimpled the machine. Natural selection really does shape every bit of our existence!

Procedure

1. Get the LEGO MINDSTORMS kit your instructor has prepared. Make sure you have all the important components in the kit. The instructor should have a list but here is the minimum: at least one motor; a light sensor; at least one touch sensor; a yellow RCX brick (the computer and battery pack, see Fig. 1); and lots of parts such as wheels, long beams, connectors, and connecting wires that run from the RCX to the motors and sensors.

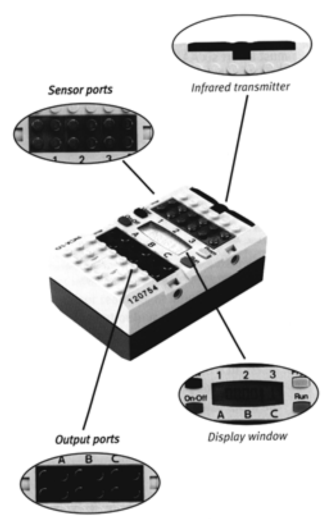
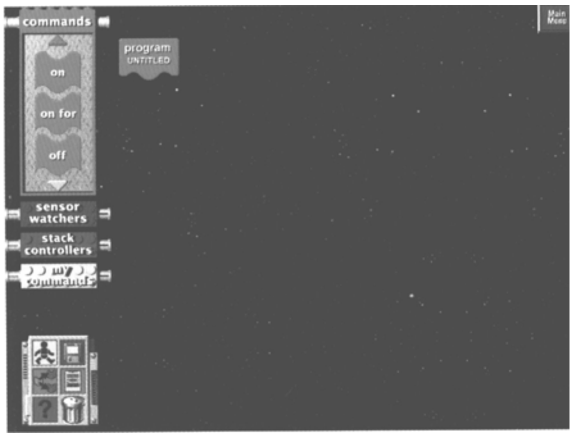


Figure 1. The RCX unit and its components.

2. Find a computer and launch the MINDSTORMS software, enter the RCX programming area. Your instructor should provide a brief introduction to the object-oriented programming language or allow you time to investigate the tutorial and LEGO manual on the software. The way the behavior of your robot is made is to assemble pre-made 'blocks' of programming (Fig. 2) and download a finished product to the RCX (Fig. 3).



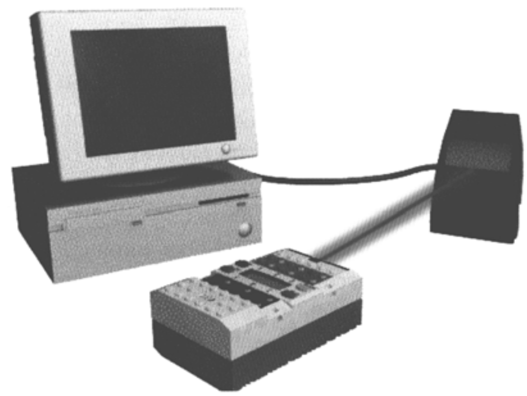
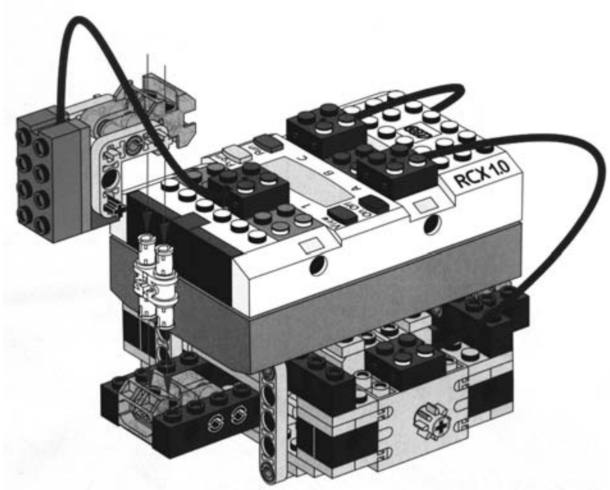


Figure 2. Screen shot of the RCX object-oriented programming window.

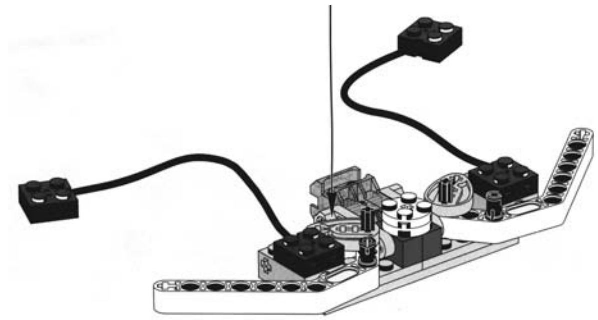
Figure 3. Infrared communication between RCX and IR transmitter hooked to PC.



3. Decide as a group what sort of creature/robot you are going to build (Fig. 4). In particular, what strategies will you follow? Do you want to use bumpers to get the robot to turn or will you rely on knowing the size and shape of the arena to lay out a search pattern? Do you want to build a slow robot or a fast one? Can the sensor detect the food patch at high speed?

Figure 4. A complete robot, try to plan your project.

4. Decide on the design of the machine. Basing your robot on one in the Constructopedia or other sources such as the MINDSTORMS website or photos from previous classes is a quicker way to go than building your own. The two crucial components are drive train and sensor placement. How are you going to drive your robot? With wheels, treads, or skids? Also, where will the motors go and how will they be anchored securely to the RCX (Fig. 5)? The motors are heavy and produce torque and they need to be strapped in. Consider the sensors: at what height does the light sensor need to be?



5. If the touch sensors are being used for bumpers, where do they need to go, and how do you build a lightweight, effective bumper (Fig. 6)?

Figure 5. Motor placement and attachment is critical, this is a Constructopedia diagram showing example placements. Notice how detailed and complex the connections are the hold the motors.

Figure 6. A Constructopedia example of a bumper design, again note complex design here, is this the best solution if low weight and maximum range of bumper coverage is the goal? This design uses both sensors, which is probably not necessary.

6. Start building and programming!

7. After your group has the motors on the robot and its basic movements programmed, start testing! The sooner the better as you want to find any problems sooner rather than later. Use the testbench function of the software to make sure the sensors are set properly. Download the program to the robot and make sure the robot moves as you mean it to.
8. Finish the assembly of the robot: sensors, bumpers, fake eyes and wings from the kit. Continue testing as you go, to try to ensure the robot is doing what you think it should.
9. Once the robot is assembled and the software is downloaded in its 'final' version, be sure to save your program on the PC. Now use the practice areas provided by the instructor to determine if your creature does what you want! One valuable concept here is that of emergent properties. Without you meaning to do so, your creature may show some patterns in its behavior. Does the combination of programmed behavior, physical parts of the robot, and the environment of the area cause the machine to make a particular pattern? This is one example of an emergent property – it is a feature of a system that arises out of the operation of the system that was not designed into the system.
10. Once you have all systems ready and the back bumper is no longer falling off every time it brushes a wall, you are ready to rumble!!!!
11. Test your robot against others in class. Each robot will be allowed to rove the simulated landscape, beeping each time its sensor detects a food source. Two runs will be done with each robot. A short period will be allowed between runs for last minute improvements and changes in your robots. The number of hits for each run will be averaged.
12. Discuss the results as a class. Why did certain robots do well and others do poorly? Did a failed robot's strategy gamble on the existence of a certain resource distribution pattern? This has direct biological implications in areas such as species invasions, community structure, and evolution of behavior. Was the cause for poor performance inadequate design of the robot or its programming? What is the biological analog of this type of problem? What other problems were seen in the class? Discuss what made the robots that did well so successful. What was it about their design and programming that worked well? What are the biological lessons that can be drawn from this?

Lab Report

Your group should prepare a report detailing your work on the project. Discuss the choices you made about constructing the robot and programming the software. Relate your decisions to the foraging strategies you used and how animals may make similar or different choices than you did.

Describe problems and successes you had during the construction of your robot. Include especially trade-offs you had to make that might be true for animals as well. Finally, describe the arena runs your robot had. How did your creature perform? What were the successes and failures of your final project? What would you have fixed if you could have? What biological animals have similar strengths and weaknesses to your robot?

The following table should help conceptualize the building process logically and biologically (See Table 1). This table, or elements of it, could be included in any report.

Table 1. Conceptual Framework for LEGO robot foraging exercise.

Decision Point	Problem	LEGO Solution	Biological Analog	Your solutions
Foraging challenge	Maximize points discovered	Construct efficient robot	Natural selection for maximum caloric intake	
Robot design	Arrangement of parts with given materials and programming	Model from Constructopedia diagrams and the create	Adaptation with phylogenic constraints and physiological trade-offs	
Placement of sensory and motors	Location of critical components is important	Basic robot types and suggested building patterns	Cephalization, bilateral symmetry, segmentation	
Optimization after assembly	Produce most efficient operation	Testbench function, practice runs	Variation by mutation and meiosis then natural selection	
Competition with other ‘bots	Outperform other robots	Your intelligence	Removal of individuals by natural selection	
Improvement period between trial runs	Effective tweaking	Your intelligence and superglue	Descent with modification	

Acknowledgements

I would like to thank students from four semesters of “Biology 490 – Evolution” for their excitement, intelligence, and their teaching of me. Also, I thank Dr. W. Wyatt Hoback for pushing me to propose this exercise as an ABLE workshop.

Literature Cited

Dacta Learning Concepts, a division of the LEGO Group: *http://www.lego.com/dacta/hardware/dcpchart.asp*

LEGO MINDSTORMS® Home Page: *http://www.MINDSTORMS.com*